Distance Covariance Analysis

Benjamin R. Cowley[1] **João D. Semedo**[1] **Amin Zandvakili**[2] **Matthew A. Smith**[3] **Adam Kohn**[4] **Byron M. Yu**[1]
[1]Carnegie Mellon University [2]Brown University [3]University of Pittsburgh [4]Albert Einstein College of Medicine

Abstract

We propose a dimensionality reduction method to identify linear projections that capture interactions between two or more sets of variables. The method, distance covariance analysis (DCA), can detect both linear and nonlinear relationships, and can take dependent variables into account. On previous testbeds and a new testbed that systematically assesses the ability to detect both linear and nonlinear interactions, DCA performs better than or comparable to existing methods, while being one of the fastest methods. To showcase the versatility of DCA, we also applied it to three different neurophysiological datasets.

1 Introduction

We consider the problem of understanding interactions between multiple sets of variables. This problem arises, for example, when studying interactions between populations of neurons in different brain areas (Semedo et al., 2014) or between different groups of genes (Winkelmann et al., 2007). These interactions are likely nonlinear (e.g., a gating mechanism between brain areas), and can be captured by a nonlinear dimensionality reduction method, such as kernel canonical correlation analysis (KCCA) (Hardoon et al., 2004; Bach and Jordan, 2002). However, nonlinear dimensionality reduction methods have two important limitations. First, the amount of data that is collected in real-world experiments is often insufficient to sample the high-dimensional space densely enough for many of these methods (Van Der Maaten et al., 2009; Cunningham and Yu, 2014). Second, most nonlinear methods provide only a low-dimensional embedding, but do not provide a direct mapping from the low-dimensional embedding to the high-dimensional data space. As a result, it is difficult to compare the topology of different low-dimensional spaces. For these reasons, many scientific (e.g., neuroscience or genetics) studies rely on linear dimensionality reduction methods, such as principal component analysis (PCA) and canonical correlation analysis (CCA) (Witten and Tibshirani, 2009; Cunningham and Yu, 2014; Kobak et al., 2016; Cowley et al., 2016).

Recently, methods that identify linear projections have been developed that can detect both linear and nonlinear interactions for multiple sets of variables. For example, hsic-CCA (Chang et al., 2013) maximizes a kernel-based correlational statistic called the Hilbert-Schmidt independence criterion (HSIC) (Gretton et al., 2007), and is a hybrid between CCA, which identifies linear projections but can only detect linear interactions, and KCCA, which can detect both linear and nonlinear interactions but identifies nonlinear projections. Thus, hsic-CCA has the interpretability of CCA as well as KCCA's ability to detect nonlinear relationships. Still, hsic-CCA is limited to identifying dimensions for two sets of variables, and cannot be used to identify dimensions for three or more sets of variables.

In this work, we propose distance covariance analysis (DCA), a dimensionality reduction method to identify linear projections that maximize the Euclidean-based correlational statistic distance covariance (Székely and Rizzo, 2009). As with HSIC, distance covariance can detect linear and nonlinear relationships (Sejdinovic et al., 2013). DCA has several important advantages over existing linear methods that can detect both linear and nonlinear relationships, such as hsic-CCA. First, DCA can identify dimensions for more than two sets of variables. Second, DCA can take into account dependent variables for which dimensions are not identified. Finally, DCA is computationally fast—in some cases, orders of magnitude faster than competing methods—without sacrificing performance. DCA can be applied to continuous and categorical variables, order the identified dimensions based on the strength of interaction, and scale to many variables and samples. Using simulated data for one, two, and multiple sets of variables, we found that DCA performed better than or comparable to existing methods, while being one of the fastest methods. We then applied DCA to real data in three different neuroscientific contexts.

2 Distance covariance

Distance covariance is a statistic that tests for independence between paired random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$, and can detect linear and nonlinear relationships between X and Y (Székely and Rizzo, 2009). The intuition is that

Proceedings of the 20^{th} International Conference on Artificial Intelligence and Statistics (AISTATS) 2017, Fort Lauderdale, Florida, USA. JMLR: W&CP volume 54. Copyright 2017 by the author(s).

if there exists a relationship between X and Y, then for two similar samples $X_i, X_j \in \mathbb{R}^p$, the two corresponding samples $Y_i, Y_j \in \mathbb{R}^q$ should also be similar. In other words, the Euclidean distance between X_i and X_j covaries with that of Y_i and Y_j. To compute the sample distance covariance $\nu(X,Y)$ for N samples, one first computes the $N \times N$ distance matrices D^X and D^Y for X and Y, respectively, where $D^X_{ij} = \|X_i - X_j\|_2$ for $i,j = 1,\ldots,N$. D^Y is computed in a similar manner. To take the covariance between distance matrices D^X and D^Y, the row, column, and matrix means must all be zero. This is achieved by computing the re-centered distance matrix R^X, where $R^X_{ij} = D^X_{ij} - \bar{D}^X_{\cdot j} - \bar{D}^X_{i\cdot} + \bar{D}^X_{\cdot\cdot}$ and the $\bar{D}^X$ terms are the row, column, and matrix means. R^Y is defined in a similar manner. The (squared) distance covariance $\nu^2(X,Y)$, a scalar, is then computed as:

$$\nu^2(X,Y) = \frac{1}{N^2} \sum_{i,j=1}^{N} R^X_{ij} R^Y_{ij} \tag{1}$$

If $\nu = 0$, then X and Y are independent (Székely and Rizzo, 2009). The formulation of distance covariance utilizes both small and large Euclidean distances, in contrast to the locality assumption of many nonlinear dimensionality reduction methods (Van Der Maaten et al., 2009). Whereas nonlinear dimensionality reduction methods seek to identify a nonlinear manifold, distance covariance seeks to detect relationships between sets of variables.

3 Optimization framework for DCA

In this section, we first formulate the DCA optimization problem for identifying a dimension[1] of X that has the greatest distance covariance with Y. We then extend this formulation for multiple sets of variables by identifying a dimension for each set that has the greatest distance covariance with all other sets. In Section 4, we propose the DCA algorithm to identify orthonormal linear dimensions ordered by decreasing distance covariance for each set of variables.

3.1 Identifying a DCA dimension for one set of variables

Consider maximizing the distance covariance between $\mathbf{u}^T X$ and Y with respect to dimension $\mathbf{u} \in \mathbb{R}^p$. We compute the (squared) distance covariance $\nu^2(\mathbf{u}^T X, Y)$ defined in (1), with re-centered distance matrix $R^X(\mathbf{u})$ for $\mathbf{u}^T X$. The optimization problem is:

$$\max_{\substack{\mathbf{u} \\ \|\mathbf{u}\| \leq 1}} \frac{1}{N^2} \sum_{i,j=1}^{N} R^Y_{ij} R^X_{ij}(\mathbf{u}) \tag{2}$$

This problem was proposed in Sheng and Yin (2013), where it was proven that the optimal solution $\mathbf{u}^*$ is a consistent estimator of $\beta \in \mathbb{R}^p$ such that X is independent of Y given $\beta^T X$. Similar guarantees exist for HSIC-related methods, such as kernel dimensionality reduction (KDR) (Fukumizu et al., 2004). However, Sheng and Yin (2013) only considered the case of identifying one dimension for one set of variables, and optimized with an approximate-gradient method (Matlab's `fmincon`).

Instead, we optimize this problem using projected gradient descent with backtracking line search. The gradient of the objective function with respect to $\mathbf{u}$ is:

$$\frac{\partial \nu^2}{d\mathbf{u}} = \frac{1}{N^2} \sum_{i,j=1}^{N} R^Y_{ij}(\delta_{ij}(\mathbf{u}) - \bar{\delta}_{\cdot j}(\mathbf{u}) - \bar{\delta}_{i\cdot}(\mathbf{u}) + \bar{\delta}_{\cdot\cdot}(\mathbf{u})) \tag{3}$$

where $\delta_{ij}(\mathbf{u}) = (X_i - X_j)\,\text{sign}(\mathbf{u}^T(X_i - X_j))$ and the $\bar{\delta}$ terms are the derivatives of the row, column, and matrix means that are used to re-center the distance matrix. For large numbers of samples, we can also make use of the fact that each gradient step is computationally inexpensive to employ stochastic projected gradient descent (with a momentum term (Hu et al., 2009) and a decaying learning rate $\tau = 0.9$).

We found that projected gradient descent performed better and was faster than other optimization approaches, such as Stiefel manifold optimization (Cunningham and Ghahramani, 2015). This is likely the case because we only optimize one dimension at a time, and do not optimize directly for multiple dimensions (see Section 4).

3.2 Identifying DCA dimensions for multiple sets of variables

Consider identifying dimensions $\mathbf{u}_1 \in \mathbb{R}^{p_1}$ and $\mathbf{u}_2 \in \mathbb{R}^{p_2}$ for two sets of variables $X^1 \in \mathbb{R}^{p_1}$ and $X^2 \in \mathbb{R}^{p_2}$, where p_1 need not equal p_2, that maximize the distance covariance by extending (2):

$$\max_{\substack{\mathbf{u}^1, \mathbf{u}^2 \\ \|\mathbf{u}^1\|, \|\mathbf{u}^2\| \leq 1}} \frac{1}{N^2} \sum_{i,j=1}^{N} R^1_{ij}(\mathbf{u}^1) R^2_{ij}(\mathbf{u}^2) \tag{4}$$

To optimize, we alternate optimizing $\mathbf{u}^1$ and $\mathbf{u}^2$, whereby on each iteration we first fix $\mathbf{u}^2$ and optimize for $\mathbf{u}^1$, then fix $\mathbf{u}^1$ and optimize for $\mathbf{u}^2$. Because of the symmetry of the objective function, each alternating optimization reduces to solving (2).

To identify dimensions for multiple sets of variables, we extend the definition of distance covariance in (1) to capture pairwise dataset interactions across M sets of variables $X^1, \ldots, X^M$ (with $X^m \in \mathbb{R}^{p_m}$), where each set may contain a different number of variables:

$$\nu(X^1, \ldots, X^M) = \frac{1}{\binom{M}{2}} \sum_{1 \leq m < n \leq M} \nu(X^m, X^n) \tag{5}$$

[1] In this work, we use the phrase "linear dimensions" or "dimensions" of a random vector $X \in \mathbb{R}^p$ to refer to either a set of orthonormal basis vectors that define a subspace in $\mathbb{R}^p$, or the projection of X onto those vectors, depending on the context.

Using (5), we extend the optimization problem of (4) to multiple sets of variables, where we desire one dimension for each set of variables that maximizes the distance covariance. We can also include information from Q sets of dependent variables $Y^1, \dots, Y^Q$ for which we are not interested in identifying dimensions but are interested in detecting their relationship with dimensions of each X^m. An example of this is a neuroscientific experiment where we identify dimensions of the recorded activity of neurons that are related across M subjects ($X^1, \dots, X^M$) *and* related to both stimulus (Y^1) and behavioral (Y^2) information.

We seek to identify dimensions $\mathbf{u}^1, \dots, \mathbf{u}^M$, where $\mathbf{u}^m \in \mathbb{R}^{p_m}$, that maximize the distance covariance $\nu({\mathbf{u}^1}^T X^1, \dots, {\mathbf{u}^M}^T X^M, Y^1, \dots, Y^Q)$. Using (5), the optimization problem is:

$$\max_{\substack{\mathbf{u}^1,\dots,\mathbf{u}^M \\ \|\mathbf{u}^m\| \leq 1}} \frac{1}{\binom{M}{2}} \frac{1}{N^2} \sum_{1 \leq m < n \leq M} \langle R^m(\mathbf{u}^m), R^n(\mathbf{u}^n) \rangle + \frac{1}{M} \frac{1}{N^2} \sum_{m=1}^{M} \langle R^m(\mathbf{u}^m), R^D \rangle \quad (6)$$

where $R^m(\mathbf{u}^m)$ is the re-centered distance matrix of ${\mathbf{u}^m}^T X^m$, $\langle R^m, R^n \rangle = \sum_{ij} R^m_{ij} R^n_{ij}$, and $R^D = \frac{1}{Q} \sum_{q=1}^{Q} R^q$ (i.e., the average of the re-centered distance matrices $R^1, \dots, R^Q$ of the sets of dependent variables $Y^1, \dots, Y^Q$). The first term is the distance covariance for multiple sets of variables as defined in (5), and the second term is the distance covariance between each set of variables and the sets of dependent variables. Similar to optimizing for two sets of variables, we optimize each $\mathbf{u}^m$ in an alternating manner which reduces solving (6) to solving (2) because we only consider terms that include $\mathbf{u}^m$.

4 DCA algorithm

For the optimization problem (6), we identify only one dimension for each set of variables. We now present the DCA algorithm (Algorithm 1), which identifies a set of DCA dimensions ordered by decreasing distance covariance for each set of variables. Given K desired DCA dimensions, DCA identifies the kth DCA dimension $\mathbf{u}^m_k$ for each of the M datasets by iteratively optimizing $\mathbf{u}^1_k, \dots, \mathbf{u}^M_k$ in (6) until some criterion is reached (e.g., the fraction of change in the objective function for two consecutive iterations is less than some ϵ). Then, the data are projected onto the orthogonal subspace of the k previously-identified dimensions before optimizing for the $(k+1)$th DCA dimension. This ensures that all subsequently-identified dimensions are orthogonal to the previously-identified dimensions. DCA returns the identified dimensions as columns in the matrices $U^1, \dots, U^M$, where $U^m \in \mathbb{R}^{p_m \times K}$, and the corresponding ordered distance covariances $d_1, \dots, d_K$.

Algorithm 1: DCA algorithm

```
Input: $\{X^1, \dots, X^M\}$, $\{Y^1, \dots, Y^Q\}$, $K$ desired dims
Output: $\{U^1, \dots, U^M\}$, $\{d_1, \dots, d_K\}$
initialize $\{U^1, \dots, U^M\}$ randomly;
for $k = 1, \dots, K$ do
    while criterion not reached do
        for $m = 1, \dots, M$ do
            $\mathbf{u}^m_k \leftarrow$
              max $\nu({\mathbf{u}^1_k}^T X^1, \dots, {\mathbf{u}^M_k}^T X^M, Y^1, \dots, Y^Q)$
              w.r.t. $\mathbf{u}^m_k$ s.t. $\|\mathbf{u}^m_k\|_2 \leq 1$
        end
    end
    $d_k \leftarrow \nu({\mathbf{u}^1_k}^T X^1, \dots, {\mathbf{u}^M_k}^T X^M, Y^1, \dots, Y^Q)$;
    for each of the $M$ datasets, $U^m(:, k) \leftarrow \mathbf{u}^m_k / \|\mathbf{u}^m_k\|_2$;
    for each of the $M$ datasets, $X^m \leftarrow$ project $X^m$ onto
      orthogonal space of $U^m(:, 1{:}k)$;
end
```

To determine the number of DCA dimensions needed, one can test if the distance covariance of the kth dimension is significant by a permutation test. Samples are first projected onto the orthogonal space of the previously-identified $(k-1)$ dimensions, because those dimensions are not considered when optimizing the kth dimension. Then, the samples are shuffled within datasets to break any relationships across datasets. A dimension is statistically significant if its distance covariance is greater than a large percentage of the distance covariances for many shuffled runs (e.g., 95% for significance level $p = 0.05$)

5 Performance on previous testbeds

We compared the performance of DCA to existing methods on testbeds used in previous work. We first considered the setting of identifying dimensions for X that are related to Y. We replicated the testbed used for KDR (Fukumizu and Leng, 2014), which included five different relationships between X and Y, ranging from sinusoidal to a 4th-degree polynomial (Fig. 1*A*). The five simulations are labeled as “A”, “B”, “C-a”, “C-b”, and “D”, matching the labeling in Fukumizu and Leng (2014). Each simulation had 10 or 50 variables, 1,000 samples, and a ground truth β whose columns determined which dimensions of X related to Y.

We then measured performance by computing the mean prinicipal angle between β and the identified $\hat{\beta}$. Existing methods included the HSIC-based methods KDR (Fukumizu and Leng, 2014) and supervised PCA (SPCA) (Barshan et al., 2011), the distance-based method supervised distance preserving projections (SDPP) (Zhu et al., 2013), the distance covariance-based method DCOV (Sheng and Yin, 2016), and as a control, PCA. Note that DCA, which optimizes dimensions sequentially, is a statistically different method than DCOV, which optimizes for all dimensions at once. For existing methods across all testbeds in this work, we used publicly available code cited by the methods’ cor-

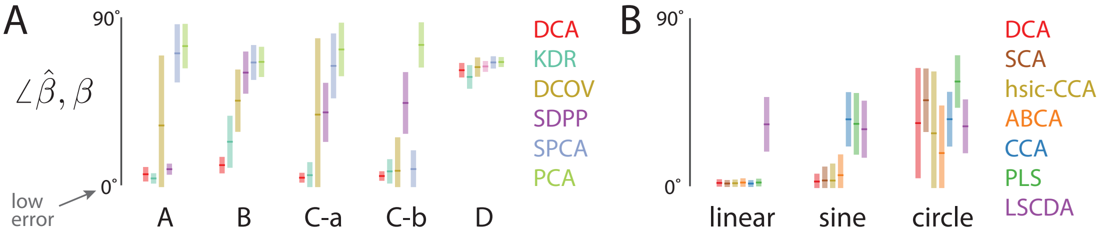

Figure 1: (*A*) KDR testbeds. (*B*) hsic-CCA testbeds. Error bars: standard deviation over 100 runs.

responding papers, as well as their suggested procedures to fit hyperparameters, such as kernel bandwidths. We found that DCA was among the best performing methods for each simulation (Fig. 1*A*, red). Simulation D showed worse performance for all methods compared to the other simulations because it required identifying 10 dimensions for 50 variables, while the other methods required identifying only a few dimensions for 10 variables.

Next, we considered the setting of identifying dimensions for both X and Y. We replicated the testbed used for hsic-CCA (Chang et al., 2013), which comprised three different relationships between variables in X (5 variables) and Y (4 variables) with 1,000 samples (Fig. 1*B*). We measured performance by computing the principal angles between the ground truth dimensions β and the identified dimensions $\hat{\beta}$ for both X and Y, and taking the average. Existing methods included those that can detect only linear interactions, such as CCA (Hotelling, 1936) and partial least squares (PLS) (Höskuldsson, 1988; Helland, 1988), as well as methods that can detect both linear and nonlinear interactions by maximizing a correlational statistic. These methods include least squares canonical dependency analysis (LSCDA, minimizing the squared-loss mutual information) (Karasuyama and Sugiyama, 2012), hsic-CCA (maximing the kernel-based HSIC statistic) (Chang et al., 2013), semiparametric canonical analysis (SCA, minimizing the prediction error of a local polynomial smoother) (Xia, 2008), and AB-canonical analysis (ABCA, maximizing the alpha-beta divergence) (Mandal and Cichocki, 2013).

Similar to the results in Fig. 1*A*, we found that DCA was among the best performing methods for the "linear" and "sine" simulations. The "circle" simulation proved challenging for all methods, with ABCA and hsic-CCA having the best mean performance (but within the error margin for DCA). The results of the two testbeds in Fig. 1 demonstrate that DCA is highly competitive with existing methods at detecting both linear and nonlinear interactions. However, these simulations tested only a small handful of linear and nonlinear relationships, and it is unclear how well these results generalize to other types of nonlinearities.

6 Performance on novel testbeds

Because the previous testbeds probed a small number of nonlinearities, we designed a testbed that allowed us to systematically vary the relationship between datasets from linear to highly nonlinear. Because existing methods are typically only applicable to one setting (identifying dimensions for one, two, or multiple sets of variables), we tested DCA separately on the three different settings for comparison. None of the existing methods can be applied to all three settings, which highlights the versatility of DCA.

6.1 Identifying dimensions for one set of variables

To systematically vary the relationship between X and Y, we generated the data (1,000 samples for each of 10 runs) as follows. Let $X = [x_1, \dots, x_{50}]^T$, where $x_i \sim \mathcal{N}(0, 1)$, and let $\beta \in \mathbb{R}^{50 \times 5}$, where each element is drawn from a standard Gaussian. The columns $\beta_1, \dots, \beta_5$ are then orthonormalized. Define each element of $Y = [y_1, \dots, y_5]^T$ as $y_i = \sin(\frac{2\pi}{\alpha} f {\beta_i}^T X)$. We chose the sine function because for $f = 1$, it is approximately linear (i.e., $\sin(x) \approx x$ for $[-\frac{\pi}{4}, \frac{\pi}{4}]$), and increases in nonlinearity with increasing f. To ensure that $\beta_i^T X$ did not exceed the domain of $[-\frac{\pi}{4}, \frac{\pi}{4}]$, we included a normalization constant $\alpha = 8\sqrt{50} \cdot \|[X_1 \ \dots \ X_{1000}]\|_\infty$. The 45 dimensions in X not related to Y represent noise, although further noise could be added to Y. We measure the performance of a method by comparing the mean principal angle between identified dimensions $\hat{\beta}$ and the ground truth β.

We tested DCA against existing methods that identified dimensions for X related to Y. We found that DCA performed well (error $< 10°$) for low frequencies, and outperformed the other methods for $60 < f < 100$ (Fig. 2*A*, top panel). DCA also ran remarkably fast— orders of magnitude faster than KDR and SPCA, which require fitting kernel bandwidths, as well as DCOV, which relies on an approximate gradient descent method (Fig. 2*A*, bottom panel). We confirmed that fitting kernel bandwidths to the data was the cause of the large computation time for KDR and SPCA by selecting a kernel bandwidth σ a priori (equal to the median of the Euclidean distances between data points). In this case, KDR-σ and SPCA-σ required similar running times as DCA (Fig. 2*A*, bottom panel). We note that for $f \leq 40$, SDPP and KDR-σ performed better than DCA with comparable running times. Thus, these methods are more appropriate for detecting linear interactions between X and Y, while DCA is more appropriate for detecting nonlinear interactions. Since DCA is a nonconvex optimization problem, we confirmed that for 100 random starts for the same X and Y (at $f = 30$), the solutions were consistent, with mean

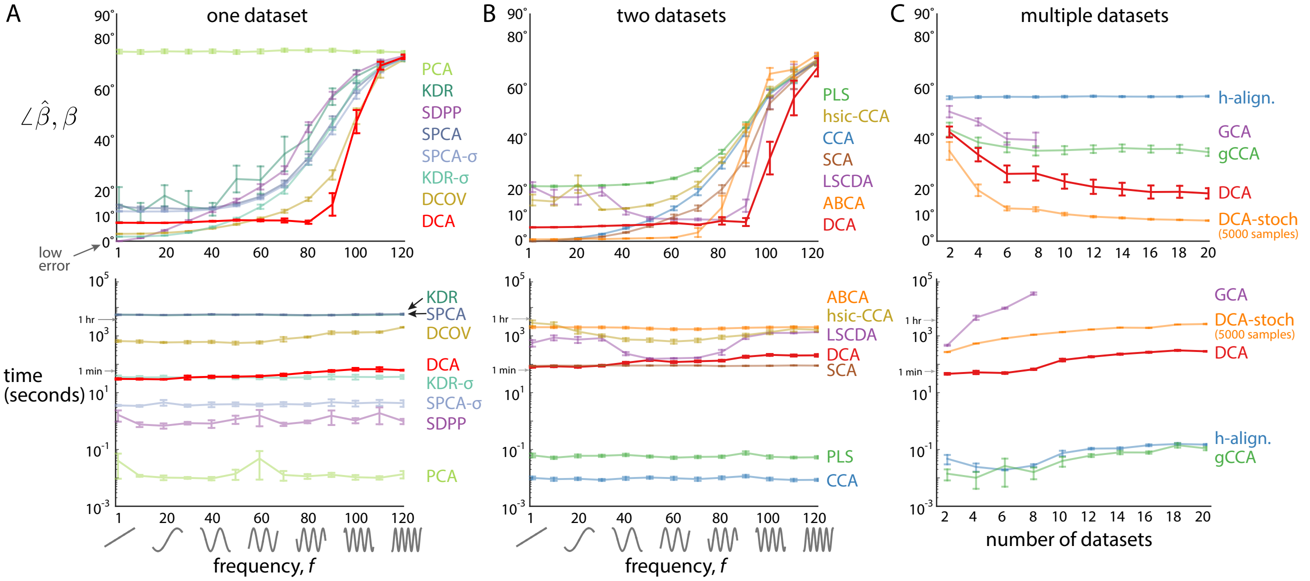

Figure 2: Top panels show error (measured by angle of overlap) for identifying dimensions for (A) one set, (B) two sets, and (C) multiple sets of variables. We varied the degree of nonlinearity in A and B, and the number of datasets in C. Bottom panels show running time in log scale. Error bars: standard deviations over 10 runs.

principal angle $7.78° \pm 0.27°$.

6.2 Identifying dimensions for two sets of variables

In the case of two sets of variables, we sought to identify dimensions for both X and Y. We used the same simulated data as in Section 6.1, and assessed how well a method recovered β, the dimensions of X related to Y (Fig. 2*B*). There are three main findings. First, DCA performed well (error $< 10°$) for low frequencies and outperformed the other methods for $80 < f < 110$. CCA, SCA, and ABCA were better able to capture linear interactions than DCA, although ABCA was much slower. We also confirmed that DCA performed well for non-orthonormalized β's and added Gaussian noise to Y (Supp. Fig. 1). Second, DCA performed better when it removed previously-identified dimensions of Y that could mask other relevant dimensions of Y (Fig. 2*B*, red curve) than when DCA did not identify dimensions of Y (Fig. 2*A*, red curve, e.g., compare with Fig. 2*B* for $f = 90$). This is an advantage shared by SCA, ABCA, and hsic-CCA (Chang et al., 2013). Finally, DCA was fast—an order of magnitude faster than some competing methods (Fig. 2*B*, bottom panel).

6.3 Identifying dimensions for multiple sets of variables

In the case of multiple sets of variables, we aimed to identify dimensions for up to $M = 20$ datasets. To test performance, we extended the previous testbed in the following way. First, we generated 500 samples of $Z \in \mathbb{R}^5$, where each element was drawn from a standard Gaussian. Then, for each set of variables $X^1, \dots, X^M \in \mathbb{R}^{10}$, we generated a random orthonormal basis $\beta^m = [\beta_1^m, \dots, \beta_5^m]$, where $\beta_i^m \in \mathbb{R}^5$. The first five of ten variables in the mth dataset $X^m = [x_1^m, \dots, x_{10}^m]^T$ were $x_i^m = \sin(\frac{2\pi}{\alpha} f {\beta_i^m}^T Z)$ for $i = 1, \dots, 5$, $m = 1, \dots, M$, $f = 30$, and $\alpha = 8\sqrt{5} \cdot \|[Z_1 \ \dots \ Z_{500}]\|_\infty$. The remaining five variables of X^m were shuffled versions of the first five variables (shuffled across samples). By generating the data in this way, we ensured that only the first five variables in each dataset were related across datasets, and we defined ground truth to be any 5-d subspace spanned by the first five variables. To measure performance, we computed the mean principal angle between the top five identified dimensions and the first five standard basis dimensions $\{e_1, \dots, e_5\}$, and averaged over the M datasets.

We found that as the number of datasets increased, the performance of most methods improved (Fig. 2*C*, top panel). This is because the methods had access to more samples to better detect interactions between datasets. DCA showed better performance than hyperalignment ("h-align.") (Haxby et al., 2011), whose PCA step returns random dimensions because all variables in X^m have equal variance. We also tested generalized CCA (gCCA) (Kettenring, 1971), which can only detect linear interactions between datasets. gCCA performed better than hyperalignment, presumably because it detected weak linear interactions between datasets, but performed worse than DCA. Finally, we tested generalized canonical analysis (GCA), a method that can detect nonlinear interactions between multiple datasets (Iaci et al., 2010), but this method performed worse and was orders of magnitude slower than DCA (Fig. 2*C*, bottom panel). We also tested the scalability of DCA by increasing the number of samples from 500 to 5,000. As expected, DCA with

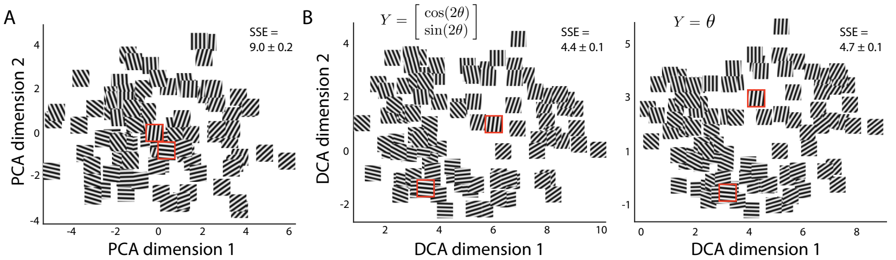

Figure 3: (*A*) PCA projection of neural responses overlaid with corresponding grating images. Red-outlined gratings have angles 90° apart. SSE: sum of squared error. (*B*) DCA projections for different Y. Same data and red-outlined gratings as in (*A*).

stochastic projected gradient descent outperformed the other methods at detecting the nonlinear relationships between datasets (Fig. 2*C*, "DCA-stoch").

7 Applications

To demonstrate how DCA can be applied to real-world data, we apply DCA to three datasets comprising recordings from tens of neurons in primary visual cortex that represent the three different settings (identifying dimensions for one, two, and multiple sets of variables). Because we do not know ground truth for these datasets, we do not compare DCA with all existing methods. Here, the purpose is to highlight how DCA returns sensible results in all three settings.

7.1 Identifying stimulus-related dimensions of neural population activity

When recording the activity of neurons in response to a sensory stimulus, there are typically aspects of the neural responses that covary with the stimulus and those that do not covary with the stimulus. Having recorded from a population of neurons, we can define a multi-dimensional activity space, where each axis represents the activity level of each neuron (Cunningham and Yu, 2014). It is of interest to identify dimensions in this space in which the population activity covaries with the stimulus (Kobak et al., 2016; Mante et al., 2013). For example, one can record from neurons in primary visual cortex (V1) and seek dimensions in which the population activity covaries with the orientation of moving bars. We analyzed a dataset in which we recorded the activity of 61 V1 neurons in response to drifting sinusoidal gratings presented for 300 ms each with the same spatial frequency and 49 different orientation angles (equally spaced between 0° and 180°) (Kelly et al., 2010). We computed the mean spike counts taken in 300 ms bins and averaged over 120 trials.

If there were a strong relationship between the neural activity and grating orientation, we would expect to see nearby neural responses encode similar grating orientations. However, when we applied PCA to the trial-averaged population activity (Fig. 3*A*), we did not observe this similarity for all nearby responses (Fig. 3*A*, red-outlined gratings). To provide supervision, we sought to identify dimensions of the trial-averaged population activity X (61 neurons × 49 orientations) that were most related to Y, the representation of grating orientation. Because we did not seek to identify dimensions in Y, this example is in the setting of identifying dimensions for one dataset.

The mean response of a V1 neuron $f(\theta)$ to different orientations θ can be described by a cosine tuning model with a preferred orientation θ_{pref} (Shriki et al., 2012). If V1 neurons were truly cosine-tuned, then $f(\theta) \propto \cos(2(\theta - \theta_{\text{pref}})) = \alpha_1 \cos(2\theta) + \alpha_2 \sin(2\theta)$, where α_1 and α_2 depend on θ_{pref}. This motivates letting $Y = [\cos(2\theta), \sin(2\theta)]^T$ to define a linear relationship between X and Y. DCA identified two dimensions in firing rate space that strongly capture orientation (Fig. 3*B*, left panel). However, if instead we represented orientation directly as $Y = \theta$ (therefore not utilizing domain knowledge), X and Y would have a nonlinear relationship. For this representation of orientation, DCA was still able to identify two dimensions that strongly capture orientation (Fig. 3*B*, right panel). For both cases, the two DCA dimensions had nearly half of the cross-validated sum of squared error (SSE), computed with linear ridge regression, than that of the two PCA dimensions. This highlights the ability of DCA to detect both linear and nonlinear relationships and return sensible results.

7.2 Identifying nonlinear relationships between neural population activity recorded in two distinct brain areas

An open question in neuroscience is how populations of neurons interact across different brain areas. Previous studies have examined linear interactions between brain areas V1 and V2 (Semedo et al., 2014). Here, we attempt to identify nonlinear interactions between V1 and V2. We analyzed a dataset in which we presented drifting sinusoidal gratings (8 orientations, each with 400 trials; 1 sec stimulus presentation for each trial) while simultaneously recording population activity from 75 V1 neurons and 22 V2 neurons (Zandvakili and Kohn, 2015). To focus on the moment-by-moment interactions, we computed the residuals of the activity by subtracting the mean spike counts from the raw spike counts (100 ms bins) for each orientation.

We asked whether DCA could identify a relationship between V1 activity and V2 activity after the linear relationship between them was removed. If so, this would imply that a nonlinear relationship exists between V1 and V2, and could be identified by DCA. We first applied CCA, PLS,

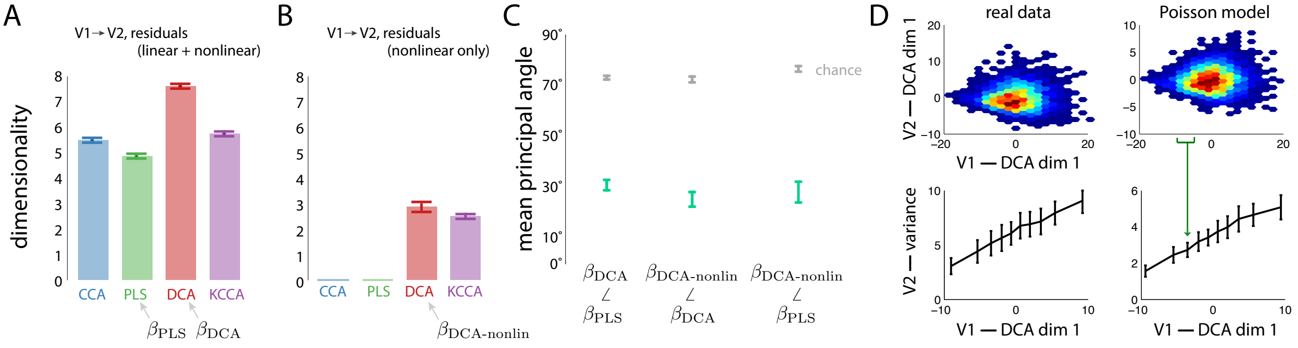

Figure 4: Dimensionality (A) between activity of V1 and V2 and (B) between the same activity, minus linear relationships. (C) Similarity between identified dimensions. (D) Top panels: Density plots of projections of the top DCA dimension of V1 and V2 activity for one grating (left, same data as in B) and for data generated from a linear-Poisson model (right). Red (blue) areas denote a high (low) density of datapoints. Bottom panels: Variance of projected V2 activity was computed in bins with equal number of datapoints; green arrow shows an example bin.

DCA, and KCCA to the activity for each orientation. We chose CCA and PLS because they can only detect linear interactions, and KCCA because it can detect nonlinear interactions. Dimensionality was determined as described in Section 4 with a significance of $p < 0.05$. We found that CCA, PLS, DCA, and KCCA all returned a dimensionality greater than zero (Fig. 4*A*), indicating that interactions exist between V1 and V2. We then subtracted the linear contribution of V1 from the V2 activity (identified with linear ridge regression). We confirmed that CCA and PLS, both linear methods, identified zero dimensions for the V2 activity that had no linear contribution from V1 (Fig. 4*B*). However, DCA identified 2 to 3 dimensions, suggesting that nonlinear interactions exist between V1 and V2. KCCA also identified a non-zero dimensionality (Fig. 4*B*), consistent with DCA.

A key advantage of methods that identify linear projections is that they can address certain types of scientific questions more readily than nonlinear methods. For example, we asked if the linear and nonlinear interactions between V1 and V2 occur along similar dimensions in the V1 activity space. It is difficult for KCCA to address this question because it does not return a mapping between the low-dimensional embedding and the high-dimensional activity space. In contrast, DCA, which identifies linear dimensions but can still detect nonlinear interactions, can readily be used to address this question. We first computed the mean principal angle between the dimensions identified by PLS and DCA in Fig. 4*A*, and found that the dimensions overlapped more than expected by chance (Fig. 4*C*, $\beta_{DCA} \angle \beta_{PLS}$, green bar lower than gray bar). This suggests that some DCA dimensions represent similar linear interactions as those identified by PLS. Next, we asked if DCA returned similar dimensions when considering the full activity (β_{DCA}, Fig. 4*A*) versus the activity minus any linear contributions ($\beta_{\text{DCA-nonlin}}$, Fig. 4*B*). As expected, the mean principal angle was small compared to chance (Fig. 4*C*, $\beta_{\text{DCA-nonlin}} \angle \beta_{\text{DCA}}$), confirming that DCA detects similar nonlinear interactions in both cases. Finally, we asked if the linear and nonlinear interactions occur along similar dimensions. We found that the mean principal angle between the PLS dimensions and the DCA-nonlinear dimensions was smaller than chance (Fig. 4*C*, $\beta_{\text{DCA-nonlin}} \angle \beta_{\text{PLS}}$). This suggests that linear and nonlinear interactions do occur along similar dimensions. Similar results hold for CCA (albeit with larger principal angles, ~60°).

To gain intuition about the nonlinear interactions identified by DCA, we plotted the top DCA dimension for V1 versus the top DCA dimension for V2 (Fig. 4*D*, top left, one representative grating). We noticed that the variance of the V2 activity increased as the V1 activity increased—a nonlinear, heteroskedastic interaction (Fig. 4*D*, bottom left). We hypothesized that a linear-nonlinear-Poisson model, where V2 activity is generated by a Poisson process whose rate is a linear projection of V1 activity passed through a hinge function, could explain this relationship. Indeed, when applying DCA to data generated from the linear-nonlinear-Poisson model, we found a similar trend as that of the real data (Fig. 4*D*, right panels).

7.3 Aligning neural population activity recorded from different subjects

The recording time (i.e., number of trials) in a given experimental session is typically limited by factors such as the subject's satiety or neural recording stability. To increase the number of trials, one can consider combining many individual datasets into one large dataset for analysis. The question is how to combine the different datasets given that possibly different neurons are recorded in each dataset. One way is to align population activity recorded from different subjects, provided that the neurons are recorded in similar brain locations. This is similar in spirit to methods that align fMRI voxels across subjects (Haxby et al., 2011). DCA

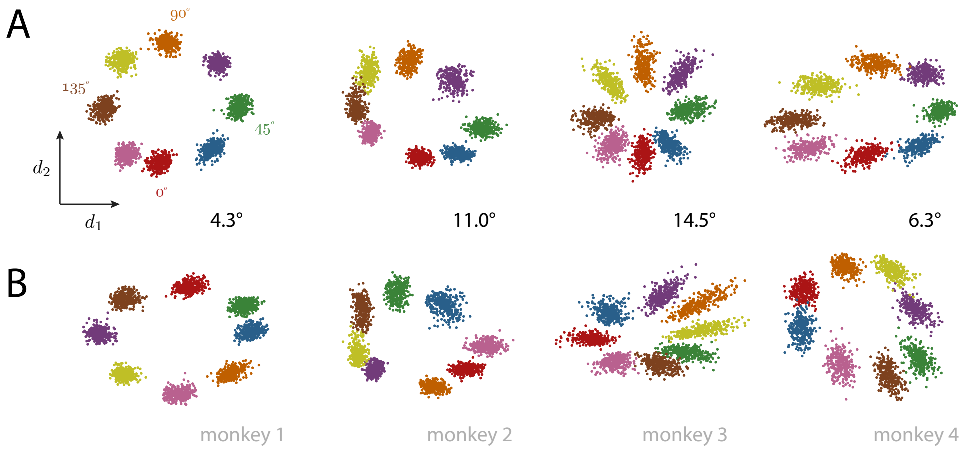

Figure 5: Top two DCA dimensions (A) of V1 population responses to gratings with 8 different orientations for 4 monkeys and (B) of the same V1 responses but with randomly permuted orientation labels. Mean principal angles were computed between dimensions in (A) and (B). Chance: ~83°.

is well-suited for alignment because of its ability to identify dimensions that are similar across subjects *and* related to stimulus information, and its ability to detect nonlinear relationships across subjects. To showcase DCA as an alignment method, we analyzed a dataset in which V1 population activity was recorded from 4 different monkeys (111, 118, 109, and 97 neurons, respectively) while drifting sinusoidal gratings (8 orientations, each with 300 trials; 1 sec stimulus presentation for each trial) were presented (Zandvakili and Kohn, 2015). We applied DCA to the 4 datasets (taken in 1 sec bins). The ith trial for each monkey corresponded to the same orientation ($i = 1, \ldots, 2400$). Given no information about orientation, DCA was able to identify dimensions of each monkey's population activity that capture orientation (Fig. 5A), because these dimensions were the most strongly related across monkeys. These dimensions were approximately linearly related because the ordering of the clusters around the circle was the same across monkeys (i.e., the DCA dimensions could be rotated to align clusters based on color).

To make the task of aligning population activity more difficult, we introduced a nonlinear transformation by randomly permuting the orientation labels across trials for each monkey. Importantly, trials that previously had the same label still had the same label after permuting (i.e., we randomly permuted the colors across clusters). After permuting, the ith trial for one monkey might not have the same orientation as the ith trial for another monkey ($i = 1, \ldots, 2400$). As before, the color labels were not provided to DCA. DCA identified remarkably similar dimensions across monkeys (Fig. 5B) as those without permutation (Fig. 5A), quantified by the mean principal angle between the dimensions (chance angle: ~83°). Because the dimensions cannot simply be rotated to align the colors of the clusters, this shows that DCA is able to detect nonlinear relationships across datasets. These results illustrate how DCA can align neural activity.

8 Discussion

We proposed DCA, a dimensionality reduction method that combines the interpretability of linear projections with the ability to detect nonlinear interactions. The biggest advantage of DCA is its applicability to a wide range of problems, including identifying dimensions in one or more sets of variables, with or without dependent variables. DCA is not regularized, unlike kernel-based methods (e.g., KDR, SPCA, and hsic-CCA), whose bandwidth parameters provide a form of regularization. However, fitting the bandwidth was computationally demanding, and when a heuristic was used to pre-select a kernel bandwidth, DCA was better able to capture nonlinear interactions. In addition, DCA may be directly regularized by the use of penalties, akin to sparse CCA (Witten and Tibshirani, 2009).

We optimized for $\{\mathbf{u}_1, \ldots, \mathbf{u}_K\}$ sequentially instead of directly optimizing for the orthonormal basis $U \in \mathbb{R}^{p \times K}$. The sequential approach, also used by other methods (Chang et al., 2013; Xia, 2008; Iaci et al., 2010; Mandal and Cichocki, 2013), has the advantages of many smaller optimization spaces rather that one large optimization space, and the removal of previously-identified dimensions that could otherwise mask other relevant dimensions (cf. Section 6.2). Directly optimizing for U no longer orders dimensions by relevance, making visualization and interpretation difficult. Still, directly optimizing for U can detect certain relationships between X and Y that would not be detected by the sequential approach (e.g., the continuous version of XOR: $Y = \text{sign}(x_1 x_2)$, where x_1 and x_2 are independent). Future work can extend DCA to directly optimize for the subspaces. The DCA source code for Matlab and Python can be found at `https://bit.ly/dca_code`.

Acknowledgments

This work was supported by NDSEG and RK Mellon Fellowships (B.R.C.), Bertucci Graduate Fellowship (J.D.S.), NIH EY022928 (M.A.S.), and Simons Collaboration on the Global Brain (A.K. and B.M.Y.).

References

F. R. Bach and M. I. Jordan. Kernel independent component analysis. *Journal of machine learning research*, 3(Jul): 1–48, 2002.

E. Barshan, A. Ghodsi, Z. Azimifar, and M. Z. Jahromi. Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds. *Pattern Recognition*, 44(7):1357–1371, 2011.

B. Chang, U. Kruger, R. Kustra, and J. Zhang. Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment. In *Proceedings of The 30th International Conference on Machine Learning*, pages 316–324, 2013.

B. R. Cowley, M. A. Smith, A. Kohn, and M. Y. Byron. Stimulus-driven population activity patterns in macaque primary visual cortex. *PLOS Computational Biology*, 12 (12):e1005185, 2016.

J. P. Cunningham and Z. Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *Journal of Machine Learning Research*, 16:2859–2900, 2015.

J. P. Cunningham and B. M. Yu. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014.

K. Fukumizu and C. Leng. Gradient-based kernel dimension reduction for regression. *Journal of the American Statistical Association*, 109(505):359–370, 2014.

K. Fukumizu, F. R. Bach, and M. I. Jordan. Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *The Journal of Machine Learning Research*, 5:73–99, 2004.

A. Gretton, K. Fukumizu, C. H. Teo, L. Song, B. Schölkopf, and A. J. Smola. A kernel statistical test of independence. In *Advances in neural information processing systems*, pages 585–592, 2007.

D. R. Hardoon, S. Szedmak, and J. Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004.

J. V. Haxby, J. S. Guntupalli, A. C. Connolly, Y. O. Halchenko, B. R. Conroy, M. I. Gobbini, M. Hanke, and P. J. Ramadge. A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, 72(2):404–416, 2011.

I. S. Helland. On the structure of partial least squares regression. *Communications in statistics-Simulation and Computation*, 17(2):581–607, 1988.

A. Höskuldsson. Pls regression methods. *Journal of chemometrics*, 2(3):211–228, 1988.

H. Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936.

C. Hu, W. Pan, and J. T. Kwok. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems*, pages 781–789, 2009.

R. Iaci, T. Sriram, and X. Yin. Multivariate association and dimension reduction: A generalization of canonical correlation analysis. *Biometrics*, 66(4):1107–1118, 2010.

M. Karasuyama and M. Sugiyama. Canonical dependency analysis based on squared-loss mutual information. *Neural Networks*, 34:46–55, 2012.

R. C. Kelly, M. A. Smith, R. E. Kass, and T. S. Lee. Local field potentials indicate network state and account for neuronal response variability. *Journal of computational neuroscience*, 29(3):567–579, 2010.

J. R. Kettenring. Canonical analysis of several sets of variables. *Biometrika*, 58(3):433–451, 1971.

D. Kobak, W. Brendel, C. Constantinidis, C. E. Feierstein, A. Kepecs, Z. F. Mainen, X.-L. Qi, R. Romo, N. Uchida, and C. K. Machens. Demixed principal component analysis of neural population data. *eLife*, 5:e10989, 2016.

A. Mandal and A. Cichocki. Non-linear canonical correlation analysis using alpha-beta divergence. *Entropy*, 15(7): 2788–2804, 2013.

V. Mante, D. Sussillo, K. V. Shenoy, and W. T. Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, 2013.

D. Sejdinovic, B. Sriperumbudur, A. Gretton, and K. Fukumizu. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5):2263–2291, 2013.

J. Semedo, A. Zandvakili, A. Kohn, C. K. Machens, and B. M. Yu. Extracting latent structure from multiple interacting neural populations. In *Advances in neural information processing systems*, pages 2942–2950, 2014.

W. Sheng and X. Yin. Direction estimation in single-index models via distance covariance. *Journal of Multivariate Analysis*, 122:148–161, 2013.

W. Sheng and X. Yin. Sufficient dimension reduction via distance covariance. *Journal of Computational and Graphical Statistics*, 25(1):91–104, 2016.

O. Shriki, A. Kohn, and M. Shamir. Fast coding of orientation in primary visual cortex. *PLoS Comput Biol*, 8(6): e1002536, 2012.

G. J. Székely and M. L. Rizzo. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009.

L. Van Der Maaten, E. Postma, and J. Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:66–71, 2009.

J. Winkelmann, B. Schormair, P. Lichtner, S. Ripke, L. Xiong, S. Jalilzadeh, S. Fulda, B. Pütz, G. Eckstein,

S. Hauk, et al. Genome-wide association study of restless legs syndrome identifies common variants in three genomic regions. *Nature genetics*, 39(8):1000–1006, 2007.

D. M. Witten and R. J. Tibshirani. Extensions of sparse canonical correlation analysis with applications to genomic data. *Statistical applications in genetics and molecular biology*, 8(1):1–27, 2009.

Y. Xia. A semiparametric approach to canonical analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(3):519–543, 2008.

A. Zandvakili and A. Kohn. Coordinated neuronal activity enhances corticocortical communication. *Neuron*, 87(4): 827–839, 2015.

Z. Zhu, T. Similä, and F. Corona. Supervised distance preserving projections. *Neural processing letters*, 38(3): 445–463, 2013.